Open. Closed. Open.

Nachum Dershowitz

School of Computer Science, Tel Aviv University
Ramat Aviv 69978, Israel
nachum.dershowitz@cs.tau.ac.il

פתוח סגור פתוח.	Open closed open.
זה כל האדם.	That is Man.

—Yehudah Amichai (Israel)
Open Closed Open (1998)

Abstract. As a window into the subject, we recount some of the history (and geography) of two mature, challenging, partially open, partially closed problems in the theory of rewriting (numbers 13 and 21 from the original *RTA List of Open Problems*). One problem deals with (criteria for left-linear) confluence and the other with termination (of one linear or string rule), the two paradigmatic properties of interest for rewrite systems of any flavor. Both problems were formulated a relatively long time ago, have seen considerable progress, but remain open. We also venture to contemplate the future evolution and impact of these investigations.

1 Introduction

Twenty years later, and we're still hitting on a keyboard.

—Michael Capellas, Chairman and CEO of Compaq (USA),
Twentieth Anniversary of the PC,
Tech Museum of Innovation (August 2001)

Rewriting—in the sense of systematically replacing symbolic terms—is as old as algebra. Diophantus of Alexandria[1] (Egypt) in his famous (ca. 3rd c. CE) book, *Arithmetica*,[2] reduced determinate and indeterminate equations to a form he knew how to solve. The use of rewriting nowadays in automated deductive engines derives from this ancient nascence of symbolic computation.

The formal study of rewriting and its properties began in 1910 with a paper by Axel Thue (Norway) [89]. Significantly, most early models of computation

[1] After whom Diophantine equations are named.

[2] It was in his copy of *this* book that Pierre de Fermat (France) wrote this frustratingly famous marginal note: *Cubem autem in duos cubos, aut quadratoquadratum in duos quadratoquadratos, et generaliter nullam in infinitum ultra quadratum potestatem in duos ejusdem nominis fas est dividere: cujus rei demonstrationem mirabilem sane detexi. Hanc marginis exiguitas non caparet.*

were based on notions of rewriting strings or terms: Thue systems [90]; Andrei Markov's (Russia) normal algorithms [66]; and Alonzo Church's (USA) lambda calculus [10]. This all led to the continued study of rewriting in the context of programming language semantics.

As a window into the history of rewriting, I have chosen two problems (numbers 13 and 21) from the original *RTA List of Open Problems* [19]. The first relates to confluence and the second, to termination.

Confluence is perhaps better known as the (equivalent) "Church-Rosser property," after a 1936 paper by Church and Barkley Rosser (USA) [11]. Properties of this property were studied shortly thereafter by Maxwell Newman (England) [73], of "Newman's Lemma" fame, and remain central in Combinatory Logic and Lambda Calculus (the immediate ancestors of the study of rewriting). Thue's paper foreshadowed the use of these and other concepts in solving word problems; see the review by Magnus Steinby (Finland) and Wolfgang Thomas (Germany) [85]. Already in 1967, Saul Gorn (USA) [37] discussed the Church-Rosser property for the use of definitions in symbolic computation.

Termination ("uniform termination" or "strong normalization") is important in automated deduction applications, to guarantee that simplification of formulæ does not itself go on forever. Simplification is often essential for reasonable performance of theorem provers. Formal proofs of termination are as old as Euclid's (Egypt) algorithm for greatest common divisor.

Gorn also did early work on proofs of termination of symbolic computation. In the abstract to his 1973 paper [36], he wrote: [3]

> This paper ... explores such questions as (1) What different interpretations can be given to the expression "the intent of the process"? (2) Does the process, or should the process end? In either event, how do we prove it? (3) If the process does end, how do we prove that it does what was intended? This question may be meaningful even if the process does not end. (4) Is there a whole class of processes that stand or fall together? Can we adapt our proof of conclusiveness to cover the whole class? (5) Do the processes of the class yield the same or different results, and whichever it is, how do we prove it?

The RTA list open problems, whence the examples herein are drawn, was created by Jan Willem Klop (The Netherlands), Jean-Pierre Jouannaud (France), and myself (USA, at the time) on the occasion of the fourth *Rewriting Techniques and Applications* conference, held in 1991 (in Italy) and chaired by the

[3] Gorn is indirectly responsible for my interest in the subject of termination of rewriting: He discussed the issue with Bob Floyd (USA), who posed a question on the subject on a 1967 qualifying exam in computer science at Carnegie-Mellon University. Zohar Manna (USA) solved the problem and went on to write a dissertation on termination. Later, Zohar showed me a 1970 paper [65] of his with Steve Ness (USA) on termination of rewriting, notes of his discussions with Steve and with Amir Pnueli (Israel) on completeness of homomorphism-based methods, as well as the dissertation of another CMU student, Renato Iturriaga (USA, at the time), thereby sparking my unquenched interest.

late Ron Book (USA).[4] Its 44 problems were compiled thanks to the contributions of many researchers who responded to messages on Pierre Lescanne's (France) rewriting mailing list,[5] and from various older lists. Updated lists subsequently appeared in the proceedings of RTA '93 (Canada) [20]—which added 33 more problems, RTA '95 (Germany) [21]—10 more, and RTA '98 (Japan) [23].

Since October 1997, the list of open problems has been maintained as a web service at

`http://www.lsv.ens-cachan.fr/~treinen/rtaloop`

This effort is spearheaded by Ralf Treinen (France). Currently, the list comprises 103 problems,[6] at least 28 of which have—gratifyingly—been solved to date, and many more have enjoyed significant progress.

2 Left-Linear Confluence

> E[lementary Problem] #1541:
> Find the maximum[7] and minimum[8] numbers of
> "Friday the 13th's" that can occur in a year.
>
> —George Clark Bush (Canada)
> *The American Mathematical Monthly* (1988)

The thirteenth problem in the original list of open problems is:

Problem #13: Give decidable (sufficient) criteria for left-linear rewriting systems to be Church-Rosser.

This problem was suggested for inclusion by Jean-Jacques Lévy (France).[9]

As already mentioned, the Church-Rosser property, $\leftrightarrow^* \subseteq \rightarrow^* {}^*\!\leftarrow$ (convergence implies joinability), had been thoroughly investigated in the context of lambda calculi and combinatory logic, and shown equivalent to the diamond confluence property ${}^*\!\leftarrow\rightarrow^* \subseteq \rightarrow^* {}^*\!\leftarrow$ (meetability implies joinability) by Max Newman (UK) in 1942 [73].[10]

---

[4] For a summary of Ron's contributions to the theory of Thue systems, a.k.a. string rewriting, see Bob McNaughton's (USA) [63]. Book and Friedrich Otto (Germany) co-authored a monograph on the subject [7].

[5] Pierre has been caretaker of this mailing list since he founded it in 1988.

[6] One more than Harvey Friedman's (USA) list of hard problems in mathematical logic in *J. Symbolic Logic* **40**(2), pp. 113–129 (1975).

[7] Three, as shown by Charles Heuer, *AMM* **70**(7), p. 759. The editors of *AMM* mistakenly asserted that there can be four if any 12-month period counts as a "year." Their retraction appeared in *AMM* **98**(7), p. 649.

[8] One or none, depending on what is meant by a "year" (*AMM*, *ibid.*).

[9] Jean-Jacques is well-known for his work on optimal strategies in the lambda calculus and for his joint work with Gérard Huet (France) on sequentiality of rewriting [47]—work that had remained in technical-report form for some 12 years.

[10] I can't help preferring $\rightarrow^*$ over $\twoheadrightarrow$ for the reflexive-transitive closure.

In 1973, Barry Rosen (USA) [81] provided a proof (albeit for the variable-free case) that shows that when a term-rewriting system is orthogonal,[11] confluence is guaranteed. In other words, when left-linear systems are also non-ambiguous (no left-hand side unifies—after renaming apart—with another left-hand side, or with any non-variable proper subterm of any left-hand side), the system is confluent. This is for much the same reason as combinatory logic is Church-Rosser, and is usually proved by recourse to an intermediate relation, such as parallel rewriting $\to^{\|}$ (for "rewriters," this means contracting redexes at disjoint, "parallel" positions), or complete developments $\to^{\perp}$(in the sense of contracting all residuals).[12]

Newman had also shown that termination plus local confluence yield the (global) confluence property.[13] Huet, in his influential 1980 paper [44], referred to the Church-Rosser property as "confluence," and provided a beautiful proof of this "Diamond Lemma," based on Noetherian (well-founded) induction.[14] Steve Kleene (USA) had given (according to Roger Hindley (UK) [41]) a simple counterexample to confluence sans termination: $\bullet \leftarrow \circ \longleftrightarrow \circ \rightarrow \bullet$. But the rewrite system for this graph has more than one rule with identical left side.

In the late 1960s, Don Knuth (USA), with a student, Peter Bendix (USA), wrote a seminal paper [52] in which they showed that confluence of critical pairs is sufficient (and necessary) for confluence of a terminating, but not necessarily left-linear, system.[15] Using the notation $s \leftarrow\!\rtimes\!\rightarrow t$ for the critical-pair relation $s = u[r]\mu \leftarrow u[l]\mu = g\mu \rightarrow d\mu = t$ (for rules $l \rightarrow r$ and $g \rightarrow d$ and most general unifier μ of l with a non-variable subterm of g in context u), this amounts to $\leftarrow\!\rtimes\!\rightarrow \subseteq \rightarrow^{*}\,{}^{*}\!\leftarrow$ (joinability, or resolvability, of critical pairs).[16] In his paper,

[11] I take some pride in having coined this term to replace its predecessors, "regular" and "non-ambiguous linear."

[12] My new notation for multi-steps at orthogonal positions.

[13] Marc Bezem (Norway) and Jan Willem Klop collect four proofs of this fact in the textbook [88] which they, and Roel de Vrijer (The Netherlands) edited: Newman's, Huet's, one based on decreasing diagrams, and one Jan Willem and I used, based on a multiset ordering of terms. For a discussion of its mechanization, see the column by Bezem and Thierry Coquand (Sweden) [5]. Unaware of Newman's lemma, several others after him proved weaker versions.

[14] This—the most general form of mathematical induction—is named after the great twentieth century algebraist, Emily Noether (Germany and the USA).

[15] Knuth is a great-great-grand-student of Thue. When I was a student, Knuth gave me an offprint of this paper (dated 1969—the conference at which it had been presented took place in 1968), since I was working on termination methods and the paper included what is now called the "Knuth-Bendix ordering."

[16] Rather than argue æsthetics, as to which way a critical pair ought to be oriented, in those cases where it matters, we use this explicit notation. Critical pairs had been presaged in a paper by Trevor Evans (USA) [24], which served as starting point for Knuth's investigations.[17] Knuth also reinvented (syntactic) unification, as used by Alan Robinson (USA) in his resolution proof procedure [80], for the purpose of calculating critical pairs, since the goal is to obtain as generic a pair as necessary to encompass all critical peaks between two rules. Bendix implemented Knuth's algorithm in Fortran.

Huet also observed that Knuth's proof of his Critical Pair Lemma does not require termination; in other words, that a system is locally confluent if, and only if, its critical pairs resolve.

Huet provided an unambiguous (critical-pair-free; hence, locally confluent) example of the necessity of left-linearity for (global) confluence of non-terminating systems: $f(x,x) \to a$, $c \to g(c)$, $f(x,g(x)) \to b$. Klop gave a similar one (with only one non-left-linear rule, but two non-terminating ones) in his foundational study [51]: $f(x,x) \to a$, $c \to g(c)$, $g(x) \to f(x,g(x))$. Six years later, my student, Sivakumar (USA, at the time) constructed the following (weakly) normalizing (every term has a normal form) and unambiguous example of non-confluence: $f(x,x) \to g(x)$, $f(x,g(x)) \to b$, $h(c,y) \to f(h(y,c),h(y,y))$.

So, confluence is decidable for (finite) terminating systems, by the Critical Pair criterion. It is, however, undecidable for non-terminating systems, since the uniform word problem is, in general, undecidable, even for string (semi-Thue) rewrite systems (see below).

The question that now begged asking was how—notwithstanding the above—one might establish the confluence of ambiguous (overlapping) non-terminating systems. Indeed, functional programmers love to write interpreters and to use streams, but still desire unique normal forms. Though they are usually content with left-linear rule patterns, it is quite natural to code nondeterministically, with ambiguous left sides.[18]

Accordingly, Huet proved that term-rewriting systems that are linear (that is, both left- and right-linear) are confluent if, but not only if, the two sides of every critical pair reduce in at most one step to a term reachable from the other side. In symbols: $\leftarrow\rtimes\rightarrow \;\subseteq\; (\rightarrow^{=}\,{}^{*}\!\leftarrow) \cap (\rightarrow^{*}\,{}^{=}\!\leftarrow)$ implies confluence. Huet also included a counterexample of Lévy's, showing the necessity for right-linearity. This criterion, however, is not very useful, since right-linearity is usually an impractical constraint, except in the string-rewriting setting (see the next section).

In any case, one cannot hope for a decidable necessary and sufficient critical-pair criterion in the general non-terminating linear case.

It was always clear that trivial critical pairs (of what are called "weakly orthogonal" systems: $\leftarrow\rtimes\rightarrow \;\subseteq\; =$) do no harm—vis-à-vis confluence, at least. Huet also proved that, without regard to right-linearity, left-linear systems are confluent if, but not only if, $\leftarrow\rtimes\rightarrow \;\subseteq\; \rightarrow^{\parallel}$, a property he dubbed "parallel closed." But his proof only works when the resolving parallel step applies to the reduct of the lower diverging step (on the open side of the symbol $\rtimes$).

Several years subsequent, in 1988, Yoshihito Toyama (Japan) [92] relaxed this condition to allow a resolution of the weak form $\rightarrow^{\parallel}\,{}^{\parallel}\!\leftarrow$, but only for critical pairs generated from two rules overlapping at their roots, a situation that we will capture with a symmetric symbol: $\leftarrow\!\bowtie\!\rightarrow$. More precisely, Toyama's sufficient condition is: $\leftarrow\rtimes\rightarrow \;\subseteq\; \rightarrow^{\parallel} \,\cup\, \big([\rightarrow^{\parallel}\,{}^{*}\!\leftarrow] \cap [\rightarrow^{*}\,{}^{\parallel}\!\leftarrow] \cap [\leftarrow\!\bowtie\!\rightarrow]\big)$. In other words, root overlay pairs need only satisfy the weaker requirement $[\rightarrow^{\parallel}\,{}^{*}\!\leftarrow] \cap [\rightarrow^{*}\,{}^{\parallel}\!\leftarrow]$.

[18] Whether non-terminating systems are necessary in the more general framework of logic programming is a question; compare my arguments in [16].

These results and many others are usually based on strong versions of local confluence, for which all one-step divergences can be resolved by some variant of rewriting for which both terms resolve in at most one step.[19] But Huet's work left open various alternative conditions on critical pairs:[20]

Problem #13a: Is $\leftarrow\rtimes\rightarrow \subseteq {}^{\|}\!\leftarrow$ also enough for confluence?

Problem #13b: If yes, then maybe some critical pairs may resolve with a step in this direction (${}^{\|}\!\leftarrow$), and others the other way around ($\rightarrow^{\|}$)? In other words: Is $\leftarrow\rtimes\rightarrow \subseteq \leftrightarrow^{\|}$ enough (where the intent is the symmetric closure of $\rightarrow^{\|}$)?

Problem #13c: If not, then what about a stronger condition, namely, $\leftarrow\rtimes\rightarrow \subseteq {}^{=}\!\leftarrow$?

Problem #13d: If yes, then one could ask whether $\leftarrow\rtimes\rightarrow \subseteq \leftrightarrow^{=}$ suffices?

A positive answer to any of these would provide a new criterion for confluence, and would suggest a Knuth-Bendix–like completion procedure for potentially non-confluent systems, adding equations to ensure that the condition is satisfied. Of course, for non-right-linear systems, a resultant critical pair may be non-linear on both sides, and, hence, unorientable. On the other hand, if these conditions are insufficient, counterexamples will have to be (besides left-linear) non-right-linear, non-terminating, and overlapping. To date, none of these conjectures has succumbed to a counterexample.

In 1991, Rolf Socher-Ambrosius (Germany) [84] wrote a short report on Problem #13a, in which an arbitrary ordering of rules induces a multiset-ordering condition on the rules used to resolve critical pairs.

In 1996, Bernhard Gramlich (Germany, at the time) [39] suggested expanding the overlaps being considered to include "parallel critical pairs."[21] Parallel rewriting is a standard tool for proving confluence of orthogonal systems, since it satisfies the diamond property ${}^{\|}\!\leftarrow\rightarrow^{\|} \subseteq \rightarrow^{\|}\,{}^{\|}\!\leftarrow$. The idea is to handle critical overlaps of such parallel steps, by requiring $\leftarrow\rtimes\rightarrow \subseteq (\rightarrow^{\|}\,{}^{*}\!\leftarrow) \cap \big(\rightarrow^{*} \cup \big([\rightarrow^{*}\,{}^{\|}\!\leftarrow] \cap [\leftarrow\bowtie\rightarrow]\big)\big)$ for all ordinary critical pairs, plus ${}^{\|}\!\leftarrow\rtimes\rightarrow \subseteq \rightarrow^{*} \cup \big([\rightarrow^{\|}\,{}^{*}\!\leftarrow] \cap [\rightarrow^{*}\,{}^{\|}\!\leftarrow] \cap [\leftarrow\bowtie\rightarrow]\big)$ for all parallel pairs.

---

19 This is an opportunity to apologize for sowing confusion by defining "strong confluence" in [18] as the "subcommutative" property, namely, $\leftarrow\rightarrow \subseteq \rightarrow^{=}\,{}^{=}\!\leftarrow$, whereas Huet used the term for his weaker condition $\leftarrow\rightarrow \subseteq (\rightarrow^{=}\,{}^{*}\!\leftarrow) \cap (\rightarrow^{*}\,{}^{=}\!\leftarrow)$.

20 Problem #13b was not posed explicitly in [19], but was included, for example, by Bernhard Gramlich (Germany, at the time) in [39].

21 Gramlich generously attributes this extension of the notion of critical pairs to what underlies what are known as "critical pair criteria," as in the works of Franz Winkler and Bruno Buchberger (Austria) [93], Wolfgang Küchlin (USA, at the time) [54], Deepak Kapur (USA), Dave Musser (USA), and P. Narendran (USA) [49], and my student, Leo Bachmair (USA), and myself [1]. Around the same time, Dave Plaisted (USA) and Andrea Sattler-Klein (Germany) [79] also employed parallel critical pairs, but for other purposes.

Actually, parallel critical pairs and a related result were already present in an unpublished 1981 report in Japanese by Toyama [91]. There, the condition was the weaker inclusion $\leftarrow\!\rtimes\!\rightarrow \subseteq (\rightarrow^{\parallel}\,{}^{*}\!\leftarrow) \cap (\rightarrow^{*}\,{}^{\parallel}\!\leftarrow)$ for ordinary critical pairs, plus ${}^{\parallel}\!\leftarrow\!\rtimes\!\rightarrow \subseteq \rightarrow^{*}\,{}^{\parallel}\!\leftarrow$ for all parallel overlaps—the latter, however, subject to the extra requirement that all variables that appear in the contractum(s) of the resolving parallel step were also within the critical parallel redexes.

The next step transpired almost immediately, when Vincent van Oostrom—in discussions with Gramlich—realized that whatever can be said for parallel rewriting can also be said for developments.[22] Accordingly, he defined a development-closed criterion, improving on Toyama's 1988 weakening of Huet's 1980 parallel-closed condition, by replacing $\rightarrow^{\parallel}$ with $\rightarrow^{\perp}$ [76]. Specifically, a system is Church-Rosser if $\leftarrow\rtimes\rightarrow \subseteq \rightarrow^{\perp} \cup \big([\rightarrow^{\perp}\,{}^{*}\!\leftarrow] \cap [\rightarrow^{*}\,{}^{\perp}\!\leftarrow] \cap [\leftarrow\!\bowtie\!\rightarrow]\big)$. In the special case where the only overlaps are at the root, the condition is $\leftarrow\!\bowtie\!\rightarrow \subseteq \big([\rightarrow^{\perp}\,{}^{*}\!\leftarrow] \cap [\rightarrow^{*}\,{}^{\perp}\!\leftarrow]\big)$, which is satisfied when $\leftarrow\!\bowtie\!\rightarrow \subseteq (\rightarrow^{\perp}\,{}^{\perp}\!\leftarrow)$.

This led Aart Middeldorp (Japan, at the time) to raise the following question:

Problem #13-1: What if the critical pair reduces by an incomplete development, that is, if ${}^{\perp}\!\leftarrow\!\rtimes\!\rightarrow \subseteq \rightarrow^{\varnothing}$, where $\rightarrow^{\varnothing}$ signifies that only some of the redexes of a complete development $\rightarrow^{\perp}$ are contracted?

van Oostrom thinks the critical-pair theorem still holds, despite the fact that the invariant used in his proof for complete developments fails.[23]

Plus, we have yet another unanswered question:

Problem #13-2: Is ${}^{\perp}\!\leftarrow\!\rtimes\!\rightarrow \subseteq {}^{\perp}\!\leftarrow$ enough for confluence?

One can go further, by considering overlaps between developments. This condition, based on what I will call "orthogonal" critical pairs (but not define),[24] was presented by Satoshi Okui (Japan) at RTA '98 [74]. The conditions are: $\leftarrow\!\rtimes\!\rightarrow \subseteq (\rightarrow^{*}\,{}^{\perp}\!\leftarrow) \cap (\rightarrow^{\perp}\,{}^{*}\!\leftarrow)$ for ordinary critical pairs, plus ${}^{\perp}\!\leftarrow\!\rtimes\!\!\ltimes\!\rightarrow \subseteq \rightarrow^{*}\,{}^{\perp}\!\leftarrow$ for all orthogonal pairs. Independently, van Oostrom had obtained the same result—again, in the higher-order context. Whereas parallel reduction is a problem in the higher-order case, complete developments work nicely for both first-order and higher-order rewriting.[25] So, Okui and van Oostrom teamed up, and now have an unpublished generalization to the higher-order case.

---

[22] van Oostrom was motivated by attempts of Tobias Nipkow (Germany) and Richard Mayr (Germany, at the time) to extend Huet's condition to handle Nipkow's "higher-order pattern" rewrite systems. Some advantages of reasoning with orthogonal steps in Church-Rosser arguments had been pointed out by Masako Takahashi (Japan) in 1995 [87]. I heard Masako present her ideas at the *Toyohashi Symposium on Theoretical Computer Science* in 1990.

[23] van Oostrom: "I recall that in 1995 I came 'close' to solving it in the plane to Japan, but then we arrived, and I've never worked on it since."

[24] Instead of "simultaneous" or "multi-step" critical pairs.

[25] Vincent presented his ideas at a 1995 seminar in Munich (Germany), where he was holding a postdoctoral position with Tobias Nipkow at the time. He applied it to $\beta\eta\Omega$-reduction—see Henk Barendregt's (The Netherlands) book [3, p. 388]—with eight orthogonal-critical pairs.

Significant progress on Problem #13a was made by Michio Oyamaguchi (Japan) and Yoshikatsu Ohta (Japan) in 1997 [77, 78]. Let $\rightarrow^{\#}$ stand for $\rightarrow^{\parallel} \cup {}^{\Lambda}\!\leftarrow$, where ${}^{\Lambda}\!\leftarrow$ signifies a root-step. They require $\leftarrow \rtimes \rightarrow \; \subseteq \; {}^{\#}\!\leftarrow \cup \big([\rightarrow^{\#}\,{}^{*}\!\leftarrow] \cap [\rightarrow^{*}\,{}^{\#}\!\leftarrow] \cap [\leftarrow\!\bowtie\!\rightarrow]\big)$, but with an additional side condition on the parallel steps. The proof involves a beautiful invariant in terms of "outside in" sequences of $\rightarrow^{\parallel}$.

Lastly, five years ago, Toshimasa Matsumoto (Japan) [69] devised a new condition on the parallel resolution of ordinary critical pairs, based on Okui's work, but the extent of its applicability is unknown.

Perhaps critical pair criteria (see fn. 21), Nicolaas de Bruijn's (The Netherlands) and van Oostrom's decreasing diagrams [8, 75], and/or abstract semantic notions of criticality, as in Claude Kirchner's (France), Maria Paola Bonacina's (Italy), and my recent work [22, 6], can contribute to a fuller understanding of this fundamental problem.

3 One-Rule Termination

> If you leave it in existence and forget about it,
> all your future rewrite commands
> will be needlessly slow.
>
> —GNU Emacs Calc 2.02 Manual

Another problem on the original list was:

> **Problem #21a: Is termination of one (left- and right-) linear rule decidable?**

This problem was contributed by Max Dauchet (France), who had recently (at RTA '89) shown that left-linearity alone is insufficient for decidability. This was the culmination of a series of efforts to delineate the borders of decidability.

Richard Lipton (USA) and Lawrence Snyder (USA) had claimed in a footnote to a 1997 paper [60] that three rules suffice for undecidability of termination. As they had not responded to a request for a proof, Huet and Dallas Lankford (USA) set out, in an unpublished report [45], to find one.[26] They used a string-rewriting simulation of Turing machines, similar to that used by Ann Yasuhara (USA) in her book on Recursion Theory [95]. Thus termination of string-rewriting systems was provenly undecidable—for an unbounded number of rules.

In the summer of 1980, visiting Lévy and Huet at INRIA, I managed to encode Turing machines in two rules, one of which was non-linear. Dauchet went one giant step further, and found a way of showing undecidability for only one non-linear rule [12, 13]. Pierre Lescanne (France) in 1994 [59] redid this more naturally, by reducing the Post Correspondence Problem to this case. So the

[26] Dallas Lankford was an early player in the field, along with Mike Ballantyne (USA). Dallas was probably the first to realize, in 1975 [57], that a process like Knuth-Bendix completion, which uses oriented equations, could replace paramodulation as a means of handling equality within resolution theorem provers.

question (still unanswered) was (and is) whether termination of one linear rule is decidable.

In a recent paper [33], Alfons Geser (USA), Aart Middeldorp (Austria), Enno Ohlebusch (Germany), and Hans Zantema (The Netherlands) leave the following question unanswered:

> **Problem #21-1: Is termination decidable for one (not necessarily linear) normalizing rule?**

Geser (Germany, at the time) constructed a (overlapping) string rule that is normalizing but neither leftmost terminating nor rightmost terminating, and one that is rightmost terminating but non-terminating [27].

Most common term-rewriting termination proofs use simplification orderings, making terms always bigger than their subterms.[27] Aart Middeldorp (Japan, at the time) and Bernhard Gramlich (France, at the time) used Dauchet's trick and showed that it is also undecidable whether there exists a simplification ordering that proves termination of a single term-rewriting rule [71] (correcting a claim in [48]).

This negative answer suggested yet another problem:

> **Problem #87: Is it decidable whether a single term-rewriting rule can be proved terminating by a monotonic ordering that is total on ground terms?**

Such orderings are important in deduction engines; see, for example, the work on unfailing completion of my former student, Jieh Hsiang (USA, at the time), with Michaël Rusinowitch (France) [43], and of Leo Bachmair (USA), Dave Plaisted (USA), and myself [2]. Zantema, who posed this one-rule problem, already knew that it is undecidable for more rules [96]. A negative solution to this question was given two years later by Geser, Middeldorp, Ohlebusch, and Zantema [32].

Now, one might think that a one-rule system is nonterminating only if it is looping in the sense of deriving a term from one of its subcontexts. But, it turns out that there is a non-looping, non-terminating one-rule term system, as well as such a two-rule string system [34]. This raises the following question:

> **Problem #95: Is there a one-rule string-rewriting system that is non-terminating but also non-looping?**

A loop would be a string derivation of the form $s \to^+ usv$. Bob McNaughton (USA) [62] has conjectured that no such rule exists.

This all brings us around to a perhaps less ambitious, but long-standing open problem for the much simpler case of string rewriting:

> **Problem #21b: Is termination of one string rule decidable?**

[27] What I called "simplification orderings" in [14] are (in the fixed-arity case) the "divisibility orders" of Graham Higman (UK) [40].

This had been mentioned in my survey with Jean-Pierre Jouannuad (France) [18], and was included in the second edition of our open problem list, in 1993.

Length-decreasing rules (however many) are obviously terminating. In 1991, Anne-Cécile Caron (France) had shown that termination is undecidable for multi-rule non-length-increasing string systems [9]. But a single length-preserving rule is only nonterminating when both sides are identical. In the latter case, one may still enquire about the length of derivations, the subject of a 1985 paper by Yves Métivier (France) [70], and of yet another problem in our original list:

> **Problem #20b: What is the best bound on the length of a derivation for a one-rule length-preserving string-rewriting (semi-Thue) system? Is it quadratic in the size of the initial term, as conjectured in [70], or of order n^k (for rules of length k and input of length n) as proved there?**

Métivier had provided a lower bound of $n^2/4$, easily reached by the derivation from $b^{n/2}a^{n/2}$ for the rule $ba \to ab$. His conjecture that this was also the upper bound for a binary alphabet was proved a few years later by Alain Bertrand (France) [4]. In that paper, Bertrand floated a new combinatorial conjecture relating to the positions of the letters in the input word.[28]

String systems are confluent when no suffix of a left side is also a prefix, since that makes them orthogonal. For right-linear systems, in general, and string systems, in particular, termination of all forward closures (a subset of derivations in which only created redexes are contracted) is valid evidence of termination [15], an idea that grew out of an unpublished preliminary note [58] by Lankford and Dave Musser (USA). Moreover, when there are no left-side overlaps, the specific string-rewriting strategy (leftmost, rightmost, etc.) does not affect termination, and (weak) normalization implies termination [38, 17].

As an example of a difficult, though non-overlapping, length-increasing rule, Zantema suggested $bbaa \to aaabbb$, a problem that itself engendered a spate of interesting work by my student, Charles Hoot (USA) [17], Elias Tahhan [86] (France, at the time), Geser [25], and others. A complete classification of termination for a rule of the form $b^i a^j \to a^k b^\ell$ was presented by Geser and Zantema at RTA '95 (see [97]), which, in turn, was subsumed by the later work of Géraud Sénizergues (France) [82] and of Yuji Kobayashi, Masashi Katsura, and Kayoko Shikishima-Tsuji (all from Japan) [53], for $b^i a^j \to r$, where $r \in \{a, b\}^*$.

Geser picked up the gauntlet, obtaining partial results for single string-rewrite rules, culminating in his dissertation [29]. Rules with only one overlap had already been solved by Winfried Kurth (Germany) in his thesis [55], who also proved decidability of existence of loops of lengths 1–3 for one-rule string systems, and showed decidability for lone rules with right sides no more than six letters long [56]. Building on ideas of McNaughton [61, 62], Geser showed decid-

[28] From inception and until recently, our on-line list stated: "Rumor has it that the conjecture has been shown true."

ability for up to nine letters [26].[29] More recently, Geser [30] proved that termination is decidable for one-rule systems that have precisely one overlap between a prefix of the left and a suffix of the right and vice-versa. For fewer overlaps, this was already known. In [83], Shikishima-Tsuji, Katsura, and Kobayashi reduced the termination problem for a *confluent* overlapping rule to the non-overlapping case.

A grid rule is one in which some letter appears equally often on both sides (or diminishes). Grid rules cover all systems amenable to a total simplification ordering. Geser showed that termination is equivalent to the non-existence of loops of length one or two, which is decidable [28].

Dieter Hofbauer (Germany) and Johannes Waldmann (Germany) showed recently that string systems admitting a termination proof by the set extension (like the multiset extension, but for sets) of a symbol precedence preserve regular languages [42]. A string system is said to be match-bounded if only a finite section of a system annotated with symbol numbers can be used in any (labelled) derivation. Geser, Hofbauer, and Waldmann showed, in a series of papers, that match-bounded string systems are terminating; match-boundedness of right sides of forward closures is a stronger termination criterion; and inverse match-bounded string systems have a termination problems; see [31]. Decidability of match-boundedness is open.

Single-threaded derivations, where each pair of successive rewrites overlap, were introduced by Wojciech Moczydłowski (Poland; now in the USA) in his Masters thesis. He showed that one-rule string systems that are cannot consume all of a contractum from the right, nor from both sides, have a decidable termination problem. The second condition entails that the systems are either terminating or single-threaded, whence they can be simulated by a two-stack pushdown automaton; the first implies that one stack's size is bounded; hence, the problem is decidable. See his joint paper with Geser [72] in these proceedings.

In sum, the jury is still out on Problem #21b, one string-rule termination. Plaisted conjectured its decidability long ago; Kurth believes it is in general undecidable; McNaughton conjectures that at least the confluent case is decidable.

Turning again to the Church-Rosser property: The critical-pair test of Knuth and Bendix gave us a decision procedure for confluence of terminating systems, which, for non-terminating systems, remains undecidable. Confluence for one string rule is decidable, by the work of Celia Wrathall (USA) [94], but undecidable, even for just twelve string rules, as per Yuri Matiyasevich[30] (Russia) [67] (see [88, p. 151]), a bound that has been pared down to five by Matiyasevich and Sénizergues [68].[31] Accordingly, the 1993 list also included the following question:

[29] Geser: "My termination sieve had a bug that I only detected after finishing my habilitation thesis in 2002. As a consequence of this bug, eight additional rules remain that cannot be solved by the methods in this paper."

[30] Of "Hilbert's Tenth Problem (Diophantine equations) is undecidable" fame.

[31] Derivability (accessibility) is undecidable for three string rules [68]. It is, however, decidable for one; see Bob McNaughton's (USA) [64].

Problem #21c: Is confluence of one linear rule decidable?

There are a number of cases for which decidability has been shown regardless of the number of linear rules. Most recently: Guillem Godoy (Spain), Ashish Tiwari (USA), and Rakesh Verma (USA) have shown decidability when variables do not appear deeper than immediately below the outer function symbol [35].

To conclude, the questions raised in this section are interesting and important for demarcating the boundaries of decidability of termination and confluence. Their resolution, however, especially in the string case, seems combinatorial in nature, though some automata-based and residual-theory techniques are now entering the picture. The methods have ramifications for other decidability and complexity questions relating to semigroups and monoids (see, for example, the work of Katsura and Kobayashi with Friedrich Otto (Germany) [50]), topics of increasing interest in this bio-informatical era.

Acknowledgements.

I thank Alfons Geser (USA), Bernhard Gramlich (Austria), Yoshihito Toyama (Japan), and Vincent van Oostrom (The Netherlands) for updates, references, and suggestions. I apologize in advance for any and all sins of omission, consequences of the need to converge and terminate.

> We use a deep mathematical result
> (namely, a minor modification of Kolmogorov's
> solution to Hilbert's 13th problem)
> to explain why fundamental physical equations are of second order.
>
> —Takeshi Yamakawa (Japan) and Vladik Kreinovich (USA),
> *International Journal of Theoretical Physics* (1999)

References

1. Leo Bachmair and Nachum Dershowitz. Critical pair criteria for completion. *J. Symbolic Computation*, 6(1):1–18, 1988.
2. Leo Bachmair, Nachum Dershowitz, and David A. Plaisted. Completion without failure. In H. Aït-Kaci and M. Nivat, editors, *Resolution of Equations in Algebraic Structures*, volume 2: Rewriting Techniques, chapter 1, pages 1–30. Academic Press, New York, 1989.
3. Henk Barendregt. *The Lambda Calculus, its Syntax and Semantics*. North-Holland, Amsterdam, second edition, 1984.
4. Alain Bertrand. Sur une conjecture d'Yves Métivier. *Theoretical Computer Science*, 123(1):21–30, 1994.
5. Marc Bezem and Thierry Coquand. Newman's Lemma — a case study in proof automation and geometric logic (Logic in Computer Science Column). *Bulletin of the EATCS*, 79:86–100, 2003.
6. Maria Paola Bonacina and Nachum Dershowitz. Abstract canonical inference. *ACM Transactions on Computational Logic*. To appear.

7. Ronald V. Book and Friedrich Otto. *String-Rewriting Systems*. Springer-Verlag, New York, 1993.
8. Nicolaas Govert de Bruijn. A note on weak diamond properties. Memorandum 1978-08, Department of Mathematics, Eindhoven University of Technology, Eindhoven, The Netherlands, August 1978.
9. Anne-Cécile Caron. Linear bounded automata and rewrite systems: Influence of initial configurations on decision properties. In *Proceedings of the International Joint Conference on Theory and Practice of Software Development, volume 1: Colloquium on Trees in Algebra and Programming (Brighton, U.K.)*, volume 493 of *Lecture Notes in Computer Science*, pages 74–89, Berlin, April 1991. Springer-Verlag.
10. Alonzo Church. *The Calculi of Lambda Conversion*, volume 6 of *Ann. Mathematics Studies*. Princeton University Press, Princeton, NJ, 1941.
11. Alonzo Church and J. Barkley Rosser. Some properties of conversion. *Transactions of the American Mathematical Society*, 39:472–482, 1936.
12. Max Dauchet. Simulation of Turing machines by a left-linear rewrite rule. In Nachum Dershowitz, editor, *Rewriting Techniques and Applications*, volume 355 of *Lecture Notes in Computer Science*, pages 109–120, Chapel Hill, NC, USA, April 1989. Springer-Verlag.
13. Max Dauchet. Simulation of Turing machines by a regular rewrite rule. *Theoretical Computer Science*, 103(2):409–420, 1992.
14. Nachum Dershowitz. A note on simplification orderings. *Information Processing Letters*, 9(5):212–215, November 1979.
15. Nachum Dershowitz. Termination of linear rewriting systems. In *Proceedings of the Eighth International Colloquium on Automata, Languages and Programming (Acre, Israel)*, volume 115 of *Lecture Notes in Computer Science*, pages 448–458, Berlin, July 1981. European Association of Theoretical Computer Science, Springer-Verlag.
16. Nachum Dershowitz. Goal solving as operational semantics. In *Proceedings of the International Logic Programming Symposium (Portland, OR)*, pages 3–17, Cambridge, MA, December 1995. MIT Press.
17. Nachum Dershowitz and Charles Hoot. Natural termination. *Theoretical Computer Science*, 142(2):179–207, May 1995.
18. Nachum Dershowitz and Jean-Pierre Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B: Formal Methods and Semantics, chapter 6, pages 243–320. North-Holland, Amsterdam, 1990.
19. Nachum Dershowitz, Jean-Pierre Jouannaud, and Jan Willem Klop. Open problems in rewriting. In Ronald. V. Book, editor, *4th International Conference on Rewriting Techniques and Applications*, volume 488 of *Lecture Notes in Computer Science*, pages 445–456, Como, Italy, April 1991. Springer-Verlag.
20. Nachum Dershowitz, Jean-Pierre Jouannaud, and Jan Willem Klop. More problems in rewriting. In C. Kirchner, editor, *Proceedings of the Fifth International Conference on Rewriting Techniques and Applications, (Montreal, Canada, June 16–18, 1993)*, volume 690 of *Lecture Notes in Computer Science*, pages 468–487. Springer, 1993.
21. Nachum Dershowitz, Jean-Pierre Jouannaud, and Jan Willem Klop. Problems in rewriting III. In Jieh Hsiang, editor, *6th International Conference on Rewriting Techniques and Applications*, volume 914 of *Lecture Notes in Computer Science*, pages 457–471, Kaiserslautern, Germany, April 1995. Springer-Verlag.

22. Nachum Dershowitz and Claude Kirchner. Abstract canonical presentations. *Theoretical Computer Science*. To appear.
23. Nachum Dershowitz and Ralf Treinen. An on-line problem database. In Tobias Nipkow, editor, *Proceedings of the Ninth International Conference on Rewriting Techniques and Applications (Tsukuba, Japan)*, volume 1379 of *Lecture Notes in Computer Science*, pages 332–342. Springer, March 1998.
24. Trevor Evans. On multiplicative systems defined by generators and relations, I. *Proceedings of the Cambridge Philosophical Society*, 47:637–649, 1951.
25. Alfons Geser. A solution to Zantema's problem. Technical Report MIP-9314, Fakultät für Mathematik und Informatik, Universität Passau, 1993.
26. Alfons Geser. Termination of one-rule string rewriting systems $\ell \to r$ where $|r| \leq 9$. Technical report, Wilhelm-Schickard-Institut, Universität Tübingen, Germany, January 1998.
27. Alfons Geser. Note on normalizing, non-terminating one-rule string rewriting systems. *Theoretical Computer Science*, 243:489–498, 2000.
28. Alfons Geser. Decidability of termination of grid string rewriting rules. *SIAM J. Comput.*, 31(4):1156–1168, 2002.
29. Alfons Geser. *Is Termination Decidable for String Rewriting with Only One Rule?* Habilitation thesis, Universität Tübingen, Germany, January 2002.
30. Alfons Geser. Termination of string rewriting rules that have one pair of overlaps. In Robert Nieuwenhuis, editor, *14th International Conference on Rewriting Techniques*, volume 2706 of *Lecture Notes in Computer Science*, pages 410–423, Valencia, Spain, June 2003. Springer-Verlag.
31. Alfons Geser, Dieter Hofbauer, and Johannes Waldmann. Termination proofs for string rewriting systems via inverse match-bounds. *J. Automat. Reason.* To appear.
32. Alfons Geser, Aart Middeldorp, Enno Ohlebusch, and Hans Zantema. Relative undecidability in the termination hierarchy of single rewrite rules. In Michel Bidoit and Max Dauchet, editors, *Theory and Practice of Software Development*, volume 1214 of *Lecture Notes in Computer Science*, pages 237–248, Lille, France, April 1997. Springer-Verlag.
33. Alfons Geser, Aart Middeldorp, Enno Ohlebusch, and Hans Zantema. Relative undecidability in term rewriting, Part I: The termination hierarchy. *Inform. and Computation*, 178(1):101–131, 2002.
34. Alfons Geser and Hans Zantema. Non-looping string rewriting. *Theoret. Informatics Appl.*, 33(3):279–301, 1999.
35. Guillem Godoy, Ashish Tiwari, and Rakesh Verma. On the confluence of linear shallow term rewrite systems. In *Proc. of the 20th Annual Symposium on Theoretical Aspects of Computer Science (Berlin, Germany)*, volume 2607 of *Lecture Notes in Computer Science*, pages 85–96, January 2003.
36. S. Gorn. On the conclusive validation of symbol manipulation processes (how do you know it has to work?). *J. of the Franklin Institute*, 296(6):499–518, December 1973.
37. Saul Gorn. Handling the growth by definition of mechanical languages. In *Proceedings of the Spring Joint Computer Conference*, pages 213–224, Philadelphia, PA, Spring 1967.
38. Bernhard Gramlich. Relating innermost, weak, uniform and modular termination of term rewriting systems. SEKI-Report SR-93-09, Fachbereich Informatik, Universität Kaiserslautern, Kaiserslautern, West Germany, 1993.
39. Bernhard Gramlich. Confluence without termination via parallel critical pairs. In *Colloquium on Trees in Algebra and Programming*, pages 211–225, 1996.

40. Graham Higman. Ordering by divisibility in abstract algebras. *Proceedings of the London Mathematical Society (3)*, 2(7):326–336, September 1952.
41. J. Roger Hindley. *The Church-Rosser Property and a Result in Combinatory Logic*. PhD thesis, University of Newcastle-upon-Tyne, 1964.
42. Dieter Hofbauer and Johannes Waldmann. Deleting string rewriting systems preserve regularity. *Theoretical Computer Science*. To appear.
43. Jieh Hsiang and Michaël Rusinowitch. On word problems in equational theories. In T. Ottmann, editor, *Proceedings of the Fourteenth EATCS International Conference on Automata, Languages and Programming (Karlsruhe, West Germany)*, volume 267 of *Lecture Notes in Computer Science*, pages 54–71, Berlin, July 1987. Springer-Verlag.
44. Gérard Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *J. of the Association for Computing Machinery*, 27(4):797–821, October 1980.
45. Gérard Huet and Dallas S. Lankford. On the uniform halting problem for term rewriting systems. Rapport laboria 283, Institut de Recherche en Informatique et en Automatique, Le Chesnay, France, March 1978.
46. Gérard Huet and Jean-Jacques Lévy. Call by need computations in non-ambiguous linear term rewriting systems. Rapport Laboria 359, Institut National de Recherche en Informatique et en Automatique, Le Chesnay, France, August 1979.
47. Gérard Huet and Jean-Jacques Lévy. Computations in orthogonal rewriting systems, I and II. In Jean-Louis Lassez and Gordon Plotkin, editors, *Computational Logic: Essays in Honor of Alan Robinson*, pages 395–443. The MIT Press, Cambridge, MA, 1991. This is a revision of [46].
48. Jean-Pierre Jouannaud and Hélène Kirchner. Construction d'un plus petit ordre de simplification. *RAIRO Theoretical Informatics*, 18(3):191–207, 1984.
49. Deepak Kapur, David R. Musser, and Paliath Narendran. Only prime superpositions need be considered for the Knuth-Bendix procedure. *J. Symbolic Computation*, 4:19–36, August 1988.
50. Masashi Katsura, Yuji Kobayashi, and Friedrich Otto. Undecidable properties of monoids with word problem solvable in linear time. Part II– Cross sections and homological and homotopical finiteness conditions. *Theoretical Computer Science*, 301(1–3):79–101, May 2003.
51. Jan Willem Klop. *Combinatory Reduction Systems*, volume 127 of *Mathematical Centre Tracts*. Mathematisch Centrum, Amsterdam, 1980.
52. Donald E. Knuth and P. B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, Oxford, U. K., 1970. Reprinted in *Automation of Reasoning 2*, Springer-Verlag, Berlin, pp. 342–376 (1983).
53. Yuji Kobayashi, Masashi Katsura, and Kayoko Shikishima-Tsuji. Termination and derivational complexity of confluent one-rule string rewriting systems. *Theoretical Computer Science*, 262(1/2):583–632, 2001.
54. Wolfgang Küchlin. A confluence criterion based on the generalized Newman Lemma. In B. F. Caviness, editor, *Proceedings of the European Conference on Computer Algebra (Linz, Austria)*, volume 204 of *Lecture Notes in Computer Science*, pages 390–399, Berlin, 1985. Springer-Verlag.
55. Winfried Kurth. *Termination und Konfluenz von Semi-Thue-Systems mit nur einer Regel*. PhD thesis, Technische Universitat Clausthal, Clausthal, Germany, 1990. In German.
56. Winfried Kurth. One-rule semi-Thue systems with loops of length one, two, or three. *Inform. Theéor. Appl.*, 30:355–268, 1996.

57. Dallas S. Lankford. Canonical inference. Memo ATP-32, Automatic Theorem Proving Project, University of Texas, Austin, TX, December 1975.
58. Dallas S. Lankford and David R. Musser. A finite termination criterion. May 1978.
59. Pierre Lescanne. On termination of one rule rewrite systems. *Theoretical Computer Science*, 132(1–2):395–401, 1994.
60. Richard Lipton and Lawrence Snyder. On the halting of tree replacement systems. In *Proceedings of the Conference on Theoretical Computer Science*, pages 43–46, Waterloo, Canada, August 1977.
61. Robert McNaughton. The uniform halting problem for one-rule semi-Thue systems. Technical Report 94-18, Dept. of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, August 1994. Correction, Aug. 1996.
62. Robert McNaughton. Well-behaved derivations in one-rule semi-Thue systems. Technical Report 95-15, Dept. of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, November 1995. Correction, July 1996.
63. Robert McNaughton. Contributions of Ronald V. Book to the theory of string-rewriting systems. *Theor. Comput. Sci.*, 207(1):13–23, 1998.
64. Robert McNaughton. Semi-Thue systems with an inhibitor. *Journal of Automated Reasoning*, 26(4):409–431, May 2001.
65. Zohar Manna and Steven Ness. On the termination of Markov algorithms. In *Proceedings of the Third Hawaii International Conference on System Science*, pages 789–792, Honolulu, HI, January 1970.
66. Andrei A. Markov. *Theory of Algorithms*. Moscow, 1954.
67. J. V. Matiyasevich. Simple examples of undecidable associative calculi. *Soviet Mathematics (Dokladi)*, 8(2):555–557, 1967.
68. Yuri Matiyasevich and Géraud Sénizergues. Decision problems for semi-Thue systems with a few rules. In Ed Clarke, editor, *Proceedings of the Eleventh Annual IEEE Symp. on Logic in Computer Science*, pages 523–531. IEEE Computer Society Press, July 1996.
69. Toshimasa Matsumoto. On confluence conditions of left-linear term rewriting systems. Master's thesis, School of Information Science, Japan Advanced Institute of Science and Technology, February 2000. In Japanese. English abstract is available at `http://www.jaist.ac.jp/library/thesis/is-master-2000/paper/tmatsumo/abstract.pdf`.
70. Yves Métivier. Calcul de longueurs de chaînes de réécriture dans le monoïde libre. *Theoretical Computer Science*, 35(1):71–87, January 1985.
71. Aart Middeldorp and Bernhard Gramlich. Simple termination is difficult. *Applied Algebra on Engineering, Communication and Computer Science*, 6(2):115–128, 1995.
72. Wojciech Moczydłowski, Jr. and Alfons Geser. Termination of single-threaded one-rule semi-Thue systems. In *Proc. of the Sixteenth International Conf. on Rewriting Techniques and Applications (Nara, Japan)*, Lecture Notes in Computer Science. Springer, April 2005.
73. Maxwell Herman Alexander Newman. On theories with a combinatorial definition of 'equivalence'. *Annals of Mathematics*, 43(2):223–243, 1942.
74. Satoshi Okui. Simultaneous critical pairs and Church-Rosser property. In Tobias Nipkow, editor, *9th International Conference on Rewriting Techniques and Applications*, volume 1379 of *Lecture Notes in Computer Science*, pages 2–16, Tsukuba, Japan, April 1998. Springer-Verlag.
75. Vincent van Oostrom. Confluence by decreasing diagrams. *Theoretical Computer Science*, 126(2):259–280, 1994.

76. Vincent van Oostrom. Development closed critical pairs. In *Proceedings of the Second International Workshop on Higher-Order Algebra, Logic, and Term Rewriting (Paderborn, Germany; September 1995)*, volume 1074 of *Lecture Notes in Computer Science*, pages 185–200. Springer, 1996.
77. Michio Oyamaguchi and Yoshikatsu Ohta. A new parallel closed condition for Church-Rosser of left-linear term rewriting systems. In Hubert Comon, editor, *8th International Conference on Rewriting Techniques and Applications*, volume 1232 of *Lecture Notes in Computer Science*, pages 187–201, Barcelona, Spain, June 1997. Springer-Verlag.
78. Michio Oyamaguchi and Yoshikatsu Ohta. On the open problems concerning Church-Rosser of left-linear term rewriting systems. *Trans. of IEICE*, E87-D(2):290–298, 2004.
79. David A. Plaisted and Andrea Sattler-Klein. Proof lengths for equational completion. *Information and Computation*, 125(2):154–170, 1996.
80. J. Alan Robinson. A machine-oriented logic based on the resolution principle. *J. of the Association for Computing Machinery*, 12(1):23–41, January 1965.
81. Barry Rosen. Tree-manipulating systems and Church-Rosser theorems. *J. of the Association for Computing Machinery*, 20(1):160–187, January 1973.
82. Geraud Sénizergues. On the termination problem for one-rule semi-Thue systems. In *Proceedings of the Seventh International Conference on Rewriting Techniques and Applications (New Brunswick, NJ)*, volume 1103 of *Lecture Notes in Computer Science*, pages 302–316. Springer-Verlag, July 1996.
83. Kayoko Shikishima-Tsuji, Masashi Katsura, and Yuji Kobayashi. On termination of confluent one-rule string-rewriting systems. *Inf. Process. Lett.*, 61(2):91–96, 1997.
84. Rolf Socher-Ambrosius. On the Church-Rosser property in left-linear systems. Technical Report TR 91/17, SUNY at Stony Brook, 1991.
85. Magnus Steinby and Wolfgang Thomas. Trees and term rewriting in 1910: On a paper by Axel Thue. *Bulletin of the European Association for Theoretical Computer Science*, 72:256–269, 2000.
86. Elias Tahhan Bittar. Complexité linéaire du problème de Zantema. *Comptes Rendus de l'Académie des Sciences, Paris*, 323:1201–1206, 1996.
87. Masako Takahashi. Parallel reductions in λ-calculus. *Information and Computation*, 118(1):120–127, April 1995.
88. "Terese" (M. Bezem, J. W. Klop and R. de Vrijer, eds.). *Term Rewriting Systems*. Cambridge University Press, 2002.
89. Axel Thue. Die Lösung eines Spezialfalles eines generellen logischen Problems. *Kra. Videnskabs-Selskabets Skrifter I. Mat. Nat. Kl.*, (8), 1910.
90. Axel Thue. Probleme über Veranderungen von Zeichenreihen nach gegeben Regeln. *Skr. Vid. Kristianaia I. Mat. Naturv. Klasse*, 10/34, 1914.
91. Yoshihito Toyama. On the Church-Rosser property of term rewriting systems. ECL Technical Report 17672, NTT, December 1981. (In Japanese.) English précis, "On parallel critical pairs" (Nov. 1995).
92. Yoshihito Toyama. Commutativity of term rewriting systems. In K. Fuchi and L. Kott, editors, *Programming of Future Generation Computers II*, pages 393–407. North-Holland, 1988.
93. Franz Winkler and Bruno Buchberger. A criterion for eliminating unnecessary reductions in the Knuth-Bendix algorithm. In *Proceedings of the Colloquium on Algebra, Combinatorics and Logic in Computer Science*, Györ, Hungary, September 1983.

94. Celia Wrathall. Confluence of one-rule Thue systems. In *Proceedings of the First International Workshop on Word Equations and Related Topics (Tubingen)*, volume 572 of *Lecture Notes in Computer Science*, pages 237–246, Berlin, 1990. Springer-Verlag.
95. Ann Yasuhara. *Recursive Function Theory and Logic*. Academic Press, 1971.
96. Hans Zantema. Total termination of term rewriting is undecidable. *Journal of Symbolic Computation*, 20(1):43–60, 1995.
97. Hans Zantema and Alfons Geser. A complete characterization of termination of $0^p1^q \to 1^r0^s$. *Applicable Algebra in Engineering, Communication, and Computing*, 11(1):1–25, 2000.